Knowledge reduction of dynamic covering decision information systems with immigration of more objects

Guangming Lang *
School of Mathematics and Computer Science, Changsha University of Science and Technology
Changsha, Hunan 410114, P.R. China

Abstract. In practical situations, it is of interest to investigate computing approximations of sets as an important step of knowledge reduction of dynamic covering decision information systems. In this paper, we present incremental approaches to computing the type-1 and type-2 characteristic matrices of dynamic coverings whose cardinalities increase with immigration of more objects. We also present the incremental algorithms of computing the second and sixth lower and upper approximations of sets in dynamic covering approximation spaces.

Keywords: Boolean matrice; Characteristic matrice; Dynamic covering approximation space; Dynamic covering information system; Rough set

1 Introduction

Covering-based rough set theory [53], as a powerful mathematical tool for studying covering approximation spaces, has attracted a lot of attention of researchers in various fields of sciences. Especially, various kinds of approximation operators have been proposed for covering approximation spaces. Recently, Wang et al. [44] transformed the computation of approximations of a set into products of the characteristic matrices and the characteristic function of the set. However, it paid little attention to approaches to calculating the characteristic matrices. In practice, the covering approximation space varies with time due to the characteristics of data collection, and the non-incremental approach to constructing the characteristic matrices is often very costly or even intractable in dynamic covering approximation spaces. It is necessary to present effective approaches to computing characteristic matrices of dynamic coverings.

To the best of our knowledge, researchers [2, 3, 15, 26, 27, 54, 55] have focused on computing approximations of sets. For instance, Chen et al. [2, 3] constructed approximations of sets when coarsening or refining attribute values. Li et al. [15] computed approximations in dominance-based rough sets approach under the variation of attribute set. Luo et al. [26, 27] studied dynamic maintenance of approximations in set-valued ordered decision systems under the attribute generalization and the variation of object set.

*Corresponding author. Tel./fax: +86 731 85258787, langguangming1984@126.com
.

Zhang et al. [54, 55] updated rough set approximations based on relation matrices and investigated neighborhood rough sets for dynamic data mining. These works demonstrate that incremental approaches are effective and efficient for computing approximations of sets. It motivates us to apply an incremental updating scheme to conduct approximations of sets by using characteristic matrices in dynamic covering approximation spaces, which will provide an effective approach to computing approximations of sets from the view of matrices.

The purpose of this paper is to compute approximations of sets by using incremental approaches in dynamic covering approximation spaces. First, we present incremental approaches to computing the type-1 and type-2 characteristic matrices in dynamic covering approximation spaces. We mainly focus on the situation: the variation of elements in coverings when adding and deleting objects. Furthermore, we provide incremental algorithms for constructing the second and sixth lower and upper approximations of sets based on the type-1 and type-2 characteristic matrices, respectively. We compare computation complexities of the incremental algorithms with those of non-incremental algorithms. Several examples are employed to illustrate that calculating approximations of sets is simplified greatly by utilizing the proposed approach.

The rest of this paper is organized as follows: Section 2 briefly reviews the basic concepts of covering-based rough set theory. In Section 3, we introduce incremental approaches to computing the type-1 and type-2 characteristic matrices of dynamic coverings whose cardinalities increase with immigration of more objects. In Section 4, we present incremental algorithms of calculating the second and sixth lower and upper approximations of sets by using the type-1 and type-2 characteristic matrices, respectively. We conclude the paper in Section 5.

2 Preliminaries

In this section, we briefly review some concepts of covering-based rough sets.

Definition 2.1 *[53] Let U be a finite universe of discourse, and $\mathscr{C}$ a family of subsets of U. Then $\mathscr{C}$ is called a covering of U if none of elements of $\mathscr{C}$ is empty and $\bigcup\{C|C \in \mathscr{C}\} = U$.*

If $\mathscr{C}$ is a covering of U, then $(U, \mathscr{C})$ is referred to as a covering approximation space.

Definition 2.2 *[44] Let $U = \{x_1, x_2, ..., x_n\}$ be a finite universe, and $\mathscr{C} = \{C_1, C_2, ..., C_m\}$ a covering of U. For any $X \subseteq U$, the second, fifth and sixth upper and lower approximations of X with respect to $\mathscr{C}$ are defined as*

(1) $SH_{\mathscr{C}}(X) = \bigcup\{C \in \mathscr{C}|C \cap X \neq \emptyset\}$, $SL_{\mathscr{C}}(X) = [SH_{\mathscr{C}}(X^c)]^c$;

(2) $IH_{\mathscr{C}}(X) = \bigcup\{N(x)|N(x) \cap X \neq \emptyset, x \in U\}$, $IL_{\mathscr{C}}(X) = \bigcup\{N(x)|N(x) \subseteq X, x \in U\}$;

(3) $XH_{\mathscr{C}}(X) = \{x \in U|N(x) \cap X \neq \emptyset\}$, $XL_{\mathscr{C}}(X) = \{x \in U|N(x) \subseteq X\}$, *where* $N(x) = \bigcap\{C_i|x \in C_i \in \mathscr{C}\}$.

For simplicity, we omit $\mathscr{C}$ in the following description of approximation operators.

Definition 2.3 *[44] Let $U=\{x_1,x_2,...,x_n\}$ be a finite universe, $\mathscr{C}=\{C_1,C_2,...,C_m\}$ a family of subsets of U, and $M_{\mathscr{C}}=(a_{ij})_{n\times m}$, where $a_{ij}=\left\{\begin{array}{ll} 1, & x_i\in C_j; \\ 0, & x_i\notin C_j. \end{array}\right.$ Then $M_{\mathscr{C}}$ is called a matrice representation of $\mathscr{C}$.*

Accordingly, we have the characteristic function $\mathcal{X}_X=\left[\begin{array}{cccccc} a_1 & a_2 & . & . & . & a_n \end{array}\right]^T$ for $X\subseteq U$, where $a_i=\left\{\begin{array}{ll} 1, & x_i\in X; \\ 0, & x_i\notin X. \end{array}\right.$.

Definition 2.4 *[44] Let $\mathscr{C}$ be a covering of the universe U, $A=(a_{ij})_{n\times m}$ and $B=(b_{ij})_{m\times p}$ Boolean matrices, $A\odot B=(c_{ij})_{n\times p}$, where $c_{ij}=\bigwedge_{k=1}^{m}(b_{kj}-a_{ik}+1)$. Then*

(1) $\Gamma(\mathscr{C})=M_{\mathscr{C}}\cdot M_{\mathscr{C}}^T=(d_{ij})_{n\times n}$ is called the type-1 characteristic matrice of $\mathscr{C}$, where $d_{ij}=\bigvee_{k=1}^{m}(a_{ik}\cdot a_{jk})$, and $M_{\mathscr{C}}\cdot M_{\mathscr{C}}^T$ is the boolean product of $M_{\mathscr{C}}$ and its transpose $M_{\mathscr{C}}^T$;

(2) $\prod(\mathscr{C})=M_{\mathscr{C}}\odot M_{\mathscr{C}}^T$ is referred to as the type-2 characteristic matrice of $\mathscr{C}$.

Wang et al. axiomatized two important types of covering approximation operators equivalently by using the type-1 and type-2 characteristic matrice of $\mathscr{C}$.

Definition 2.5 *[44] Let $U=\{x_1,x_2,...,x_n\}$ be a finite universe, $\mathscr{C}=\{C_1,C_2,...,C_m\}$ a covering of U, and $\mathcal{X}_X$ the characteristic function of X in U. Then*

(1) $\mathcal{X}_{SH(X)}=\Gamma(\mathscr{C})\cdot\mathcal{X}_X$, $\mathcal{X}_{SL(X)}=\Gamma(\mathscr{C})\odot\mathcal{X}_X$; (2) $\mathcal{X}_{XH(X)}=\prod(\mathscr{C})\cdot\mathcal{X}_X$, $\mathcal{X}_{XL(X)}=\prod(\mathscr{C})\odot\mathcal{X}_X$.

3 Update approximations of sets with immigration of more objects

In this section, we introduce incremental approaches to computing the second and sixth lower and upper approximation of sets with immigration of more objects.

Definition 3.1 *Let $(U,\mathscr{C})$ and $(U^+,\mathscr{C}^+)$ be covering approximation spaces, where $U=\{x_1,x_2,...,x_n\}$, $U^+=U\cup\{x_{n+1},x_{n+2},...,x_{n+t}\}(t\geq 2)$, $\mathscr{C}=\{C_1,C_2,...,C_m\}$, $\mathscr{C}^+=\{C_1^+,C_2^+,...,C_m^+,C_{m+1}^+,C_{m+2}^+,...,C_{m+l}^+\}(l\geq 2)$, where $C_i^+=C_i\cup\Delta C_i$ or C_i $(1\leq i\leq m)$, $\Delta C_i\subseteq\{x_{n+1},x_{n+2},...,x_{n+t}\}$, and $\{x_{n+1},x_{n+2},...,x_{n+t}\}\subseteq\{C_{m+j}^+|1\leq j\leq l\}$. Then $(U^+,\mathscr{C}^+)$ is called a dynamic covering approximation space.*

By Definition 3.1, we refer $\mathscr{C}^+$ to as a dynamic covering. Although there are several types of coverings when adding objects, we only discuss this type of dynamic coverings for simplicity in this work.

In what follows, we discuss how to construct $\Gamma(\mathscr{C}^+)$ based on $\Gamma(\mathscr{C})$. For convenience, we denote $M_{\mathscr{C}}=(a_{ij})_{n\times m}$, $M_{\mathscr{C}^+}=(a_{ij})_{(n+t)\times(m+l)}$, $\Gamma(\mathscr{C})=(b_{ij})_{n\times n}$ and $\Gamma(\mathscr{C}^+)=(c_{ij})_{(n+t)\times(n+t)}$.

Theorem 3.2 *Let $(U^+,\mathscr{C}^+)$ be a dynamic covering approximation space of $(U,\mathscr{C})$, $\Gamma(\mathscr{C})$ and $\Gamma(\mathscr{C}^+)$ the type-1 characteristic matrices of $\mathscr{C}$ and $\mathscr{C}^+$, respectively. Then*

$$\Gamma(\mathscr{C}^+)=\left[\begin{array}{cc} \Gamma(\mathscr{C}) & 0 \\ 0 & 0 \end{array}\right]\bigvee\left[\begin{array}{cc} \triangle_1(\Gamma(\mathscr{C})) & (\triangle_2(\Gamma(\mathscr{C})))^T \\ \triangle_2(\Gamma(\mathscr{C})) & \triangle_3(\Gamma(\mathscr{C})) \end{array}\right],$$

where

$$\triangle_1(\Gamma(\mathscr{C})) = \begin{bmatrix} a_{1(m+1)} & a_{2(m+1)} & \cdot & \cdot & \cdot & a_{n(m+1)} \\ a_{1(m+2)} & a_{2(m+2)} & \cdot & \cdot & \cdot & a_{n(m+2)} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{1(m+l)} & a_{2(m+l)} & \cdot & \cdot & \cdot & a_{n(m+l)} \end{bmatrix}^T \cdot \begin{bmatrix} a_{1(m+1)} & a_{2(m+1)} & \cdot & \cdot & \cdot & a_{n(m+1)} \\ a_{1(m+2)} & a_{2(m+2)} & \cdot & \cdot & \cdot & a_{n(m+2)} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{1(m+l)} & a_{2(m+l)} & \cdot & \cdot & \cdot & a_{n(m+l)} \end{bmatrix};$$

$$\triangle_2(\Gamma(\mathscr{C})) = \begin{bmatrix} a_{(n+1)1} & a_{(n+1)2} & \cdot & \cdot & \cdot & a_{(n+1)(m+l)} \\ a_{(n+2)1} & a_{(n+2)2} & \cdot & \cdot & \cdot & a_{(n+2)(m+l)} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{(n+t)1} & a_{(n+t)2} & \cdot & \cdot & \cdot & a_{(n+t)(m+l)} \end{bmatrix} \cdot \begin{bmatrix} a_{11} & a_{12} & \cdot & \cdot & \cdot & a_{1(m+l)} \\ a_{21} & a_{22} & \cdot & \cdot & \cdot & a_{2(m+l)} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{n1} & a_{n2} & \cdot & \cdot & \cdot & a_{n(m+l)} \end{bmatrix}^T;$$

$$\triangle_3(\Gamma(\mathscr{C})) = \begin{bmatrix} a_{(n+1)1} & a_{(n+1)2} & \cdot & \cdot & \cdot & a_{(n+1)(m+l)} \\ a_{(n+2)1} & a_{(n+2)2} & \cdot & \cdot & \cdot & a_{(n+2)(m+l)} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{(n+t)1} & a_{(n+t)2} & \cdot & \cdot & \cdot & a_{(n+t)(m+l)} \end{bmatrix} \cdot \begin{bmatrix} a_{(n+1)1} & a_{(n+1)2} & \cdot & \cdot & \cdot & a_{(n+1)(m+l)} \\ a_{(n+2)1} & a_{(n+2)2} & \cdot & \cdot & \cdot & a_{(n+2)(m+l)} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{(n+t)1} & a_{(n+t)2} & \cdot & \cdot & \cdot & a_{(n+t)(m+l)} \end{bmatrix}^T.$$

Proof. By Definition 3.1, we get $\Gamma(\mathscr{C})$ and $\Gamma(\mathscr{C}^+)$ as follows:

$$\begin{aligned} \Gamma(\mathscr{C}) &= M_{\mathscr{C}} \cdot M_{\mathscr{C}}^T \\ &= \begin{bmatrix} a_{11} & a_{12} & \cdot & \cdot & \cdot & a_{1m} \\ a_{21} & a_{22} & \cdot & \cdot & \cdot & a_{2m} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{n1} & a_{n2} & \cdot & \cdot & \cdot & a_{nm} \end{bmatrix} \cdot \begin{bmatrix} a_{11} & a_{12} & \cdot & \cdot & \cdot & a_{1m} \\ a_{21} & a_{22} & \cdot & \cdot & \cdot & a_{2m} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{n1} & a_{n2} & \cdot & \cdot & \cdot & a_{nm} \end{bmatrix}^T \\ &= \begin{bmatrix} b_{11} & b_{12} & \cdot & \cdot & \cdot & b_{1n} \\ b_{21} & b_{22} & \cdot & \cdot & \cdot & b_{2n} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ b_{n1} & b_{n2} & \cdot & \cdot & \cdot & b_{nn} \end{bmatrix}; \end{aligned}$$

$$
\begin{aligned}
\Gamma(\mathscr{C}^+) &= M_{\mathscr{C}^+} \cdot M_{\mathscr{C}^+}^T \\
&= \begin{bmatrix}
a_{11} & a_{12} & \cdot & \cdot & \cdot & a_{1m} & a_{1(m+1)} & \cdot & \cdot & \cdot & a_{1(m+l)} \\
a_{21} & a_{22} & \cdot & \cdot & \cdot & a_{2m} & a_{2(m+1)} & \cdot & \cdot & \cdot & a_{2(m+l)} \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
a_{n1} & a_{n2} & \cdot & \cdot & \cdot & a_{nm} & a_{n(m+1)} & \cdot & \cdot & \cdot & a_{n(m+l)} \\
a_{(n+1)1} & a_{(n+1)2} & \cdot & \cdot & \cdot & a_{(n+1)m} & a_{(n+1)(m+1)} & \cdot & \cdot & \cdot & a_{(n+1)(m+l)} \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
a_{(n+t)1} & a_{(n+t)2} & \cdot & \cdot & \cdot & a_{(n+t)m} & a_{(n+t)(m+1)} & \cdot & \cdot & \cdot & a_{(n+t)(m+l)}
\end{bmatrix} \\
&\quad \cdot \begin{bmatrix}
a_{11} & a_{12} & \cdot & \cdot & \cdot & a_{1m} & a_{1(m+1)} & \cdot & \cdot & \cdot & a_{1(m+l)} \\
a_{21} & a_{22} & \cdot & \cdot & \cdot & a_{2m} & a_{2(m+1)} & \cdot & \cdot & \cdot & a_{2(m+l)} \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
a_{n1} & a_{n2} & \cdot & \cdot & \cdot & a_{nm} & a_{n(m+1)} & \cdot & \cdot & \cdot & a_{n(m+l)} \\
a_{(n+1)1} & a_{(n+1)2} & \cdot & \cdot & \cdot & a_{(n+1)m} & a_{(n+1)(m+1)} & \cdot & \cdot & \cdot & a_{(n+1)(m+l)} \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
a_{(n+t)1} & a_{(n+t)2} & \cdot & \cdot & \cdot & a_{(n+t)m} & a_{(n+t)(m+1)} & \cdot & \cdot & \cdot & a_{(n+t)(m+l)}
\end{bmatrix}^T \\
&= \begin{bmatrix}
c_{11} & c_{12} & \cdot & \cdot & \cdot & c_{1n} & c_{1(n+1)} & \cdot & \cdot & \cdot & c_{1(n+t)} \\
c_{21} & c_{22} & \cdot & \cdot & \cdot & c_{2n} & c_{2(n+1)} & \cdot & \cdot & \cdot & c_{2(n+t)} \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
c_{n1} & c_{n2} & \cdot & \cdot & \cdot & c_{nn} & c_{n(n+1)} & \cdot & \cdot & \cdot & c_{n(n+t)} \\
c_{(n+1)1} & c_{(n+1)2} & \cdot & \cdot & \cdot & c_{(n+1)n} & c_{(n+1)(n+1)} & \cdot & \cdot & \cdot & c_{(n+1)(n+t)} \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\
c_{(n+t)1} & c_{(n+t)2} & \cdot & \cdot & \cdot & c_{(n+t)n} & c_{(n+t)(n+1)} & \cdot & \cdot & \cdot & c_{(n+t)(n+t)}
\end{bmatrix};
\end{aligned}
$$

In the sense of the type-1 characteristic matrice of $\mathscr{C}^+$, we have

$$
\begin{aligned}
c_{11} &= [\ a_{11} \quad a_{12} \quad . \quad . \quad . \quad a_{1m} \quad a_{1(m+1)} \quad . \quad . \quad . \quad a_{1(m+l)}\ ] \\
&\quad \cdot [\ a_{11} \quad a_{12} \quad . \quad . \quad . \quad a_{1m} \quad a_{1(m+1)} \quad . \quad . \quad . \quad a_{1(m+l)}\ ]^T \\
&= [\ a_{11} \quad a_{12} \quad . \quad . \quad . \quad a_{1m}\ ] \cdot [\ a_{11} \quad a_{12} \quad . \quad . \quad . \quad a_{1m}\ ]^T \\
&\quad \vee [\ a_{1(m+1)} \quad . \quad . \quad . \quad a_{1(m+l)}\ ] \cdot [\ a_{1(m+1)} \quad . \quad . \quad . \quad a_{1(m+l)}\ ]^T \\
&= b_{11} \vee [\ a_{1(m+1)} \quad . \quad . \quad . \quad a_{1(m+l)}\ ] \cdot [\ a_{1(m+1)} \quad . \quad . \quad . \quad a_{1(m+l)}\ ]^T; \\
c_{(n+1)1} &= [\ a_{(n+1)1} \quad a_{(n+1)2} \quad . \quad . \quad . \quad a_{(n+1)m} \quad a_{(n+1)(m+1)} \quad . \quad . \quad . \quad a_{(n+1)(m+l)}\ ] \\
&\quad \cdot [\ a_{11} \quad a_{12} \quad . \quad . \quad . \quad a_{1m} \quad a_{1(m+1)} \quad . \quad . \quad . \quad a_{1(m+l)}\ ]^T \\
&= 0 \vee [\ a_{11} \quad a_{12} \quad . \quad . \quad . \quad a_{1m} \quad a_{1(m+1)} \quad . \quad . \quad . \quad a_{1(m+l)}\ ]^T;
\end{aligned}
$$

$$\begin{aligned}
c_{(n+1)(n+1)} &= \begin{bmatrix} a_{(n+1)1} & a_{(n+1)2} & \cdot & \cdot & \cdot & a_{(n+1)m} & a_{(n+1)(m+1)} & \cdot & \cdot & \cdot & a_{(n+1)(m+l)} \end{bmatrix} \\
&\quad \cdot \begin{bmatrix} a_{(n+1)1} & a_{(n+1)2} & \cdot & \cdot & \cdot & a_{(n+1)m} & a_{(n+1)(m+1)} & \cdot & \cdot & \cdot & a_{(n+1)(m+l)} \end{bmatrix}^T \\
&= 0 \vee \begin{bmatrix} a_{(n+1)1} & a_{(n+1)2} & \cdot & \cdot & \cdot & a_{(n+1)m} & a_{(n+1)(m+1)} & \cdot & \cdot & \cdot & a_{(n+1)(m+l)} \end{bmatrix} \\
&\quad \cdot \begin{bmatrix} a_{(n+1)1} & a_{(n+1)2} & \cdot & \cdot & \cdot & a_{(n+1)m} & a_{(n+1)(m+1)} & \cdot & \cdot & \cdot & a_{(n+1)(m+l)} \end{bmatrix}^T.
\end{aligned}$$

Since $c_{11} \in \triangle_1(\Gamma(\mathscr{C}))$, $c_{(n+1)1} \in \triangle_2(\Gamma(\mathscr{C}))$ and $c_{(n+1)(n+1)} \in \triangle_3(\Gamma(\mathscr{C}))$, we can compute other elements of $\triangle_1(\Gamma(\mathscr{C}))$, $\triangle_2(\Gamma(\mathscr{C}))$ and $\triangle_3(\Gamma(\mathscr{C}))$ similarly. Thus, to obtain $\Gamma(\mathscr{C}^+)$, we only need to compute $\triangle_1(\Gamma(\mathscr{C}))$, $\triangle_2(\Gamma(\mathscr{C}))$ and $\triangle_3(\Gamma(\mathscr{C}))$ on the basis of $\Gamma(\mathscr{C})$ as follows:

$$\triangle_1(\Gamma(\mathscr{C})) = \begin{bmatrix} a_{1(m+1)} & a_{2(m+1)} & \cdot & \cdot & \cdot & a_{n(m+1)} \\ a_{1(m+2)} & a_{2(m+2)} & \cdot & \cdot & \cdot & a_{n(m+2)} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{1(m+l)} & a_{2(m+l)} & \cdot & \cdot & \cdot & a_{n(m+l)} \end{bmatrix}^T \cdot \begin{bmatrix} a_{1(m+1)} & a_{2(m+1)} & \cdot & \cdot & \cdot & a_{n(m+1)} \\ a_{1(m+2)} & a_{2(m+2)} & \cdot & \cdot & \cdot & a_{n(m+2)} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{1(m+l)} & a_{2(m+l)} & \cdot & \cdot & \cdot & a_{n(m+l)} \end{bmatrix};$$

$$\triangle_2(\Gamma(\mathscr{C})) = \begin{bmatrix} a_{(n+1)1} & a_{(n+1)2} & \cdot & \cdot & \cdot & a_{(n+1)(m+l)} \\ a_{(n+2)1} & a_{(n+2)2} & \cdot & \cdot & \cdot & a_{(n+2)(m+l)} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{(n+t)1} & a_{(n+t)2} & \cdot & \cdot & \cdot & a_{(n+t)(m+l)} \end{bmatrix} \cdot \begin{bmatrix} a_{11} & a_{12} & \cdot & \cdot & \cdot & a_{1(m+l)} \\ a_{21} & a_{22} & \cdot & \cdot & \cdot & a_{2(m+l)} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{n1} & a_{n2} & \cdot & \cdot & \cdot & a_{n(m+l)} \end{bmatrix}^T;$$

$$\triangle_3(\Gamma(\mathscr{C})) = \begin{bmatrix} a_{(n+1)1} & a_{(n+1)2} & \cdot & \cdot & \cdot & a_{(n+1)(m+l)} \\ a_{(n+2)1} & a_{(n+2)2} & \cdot & \cdot & \cdot & a_{(n+2)(m+l)} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{(n+t)1} & a_{(n+t)2} & \cdot & \cdot & \cdot & a_{(n+t)(m+l)} \end{bmatrix} \cdot \begin{bmatrix} a_{(n+1)1} & a_{(n+1)2} & \cdot & \cdot & \cdot & a_{(n+1)(m+l)} \\ a_{(n+2)1} & a_{(n+2)2} & \cdot & \cdot & \cdot & a_{(n+2)(m+l)} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ a_{(n+t)1} & a_{(n+t)2} & \cdot & \cdot & \cdot & a_{(n+t)(m+l)} \end{bmatrix}^T.$$

Therefore, we have

$$\Gamma(\mathscr{C}^+) = \begin{bmatrix} \Gamma(\mathscr{C}) & 0 \\ 0 & 0 \end{bmatrix} \bigvee \begin{bmatrix} \triangle_1(\Gamma(\mathscr{C})) & (\triangle_2(\Gamma(\mathscr{C})))^T \\ \triangle_2(\Gamma(\mathscr{C})) & \triangle_3(\Gamma(\mathscr{C})) \end{bmatrix}.$$

Example 3.3 *Let* $U = \{x_1, x_2, x_3, x_4\}$, $U^+ = U \cup \{x_5, x_6\}$, $\mathscr{C} = \{C_1, C_2, C_3\}$, $\mathscr{C}^+ = \{C_1^+, C_2^+, C_3^+, C_4^+, C_5^+\}$, *where* $C_1 = \{x_1, x_4\}$, $C_2 = \{x_1, x_2, x_4\}$, $C_3 = \{x_3, x_4\}$, $C_1^+ = \{x_1, x_4, x_5\}$, $C_2^+ = \{x_1, x_2, x_4, x_5\}$, $C_3^+ = \{x_3, x_4\}$, $C_4^+ = \{x_3, x_5, x_6\}$,$C_5^+ = \{x_1, x_6\}$, *and* $X = \{x_3, x_4, x_5\}$. *By Definition 3.1, we first have that*

$$\Gamma(\mathscr{C}) = M_{\mathscr{C}} \cdot M_{\mathscr{C}}^T = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}^T = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{bmatrix}.$$

Second, by Theorem 3.2, we get that

$$\triangle_1(\Gamma(\mathscr{C})) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}^T \cdot \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix};$$

$$\triangle_2(\Gamma(\mathscr{C})) = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 \end{bmatrix}^T = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{bmatrix};$$

$$\triangle_3(\Gamma(\mathscr{C})) = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix}^T = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Thus, we obtain that

$$\begin{aligned} \Gamma(\mathscr{C}^+) &= (c_{ij})_{66} \\ &= \begin{bmatrix} \Gamma(\mathscr{C}) & 0 \\ 0 & 0 \end{bmatrix} \bigvee \begin{bmatrix} \triangle_1(\Gamma(\mathscr{C}) & (\triangle_2(\Gamma(\mathscr{C})))^T \\ \triangle_2(\Gamma(\mathscr{C})) & \triangle_3(\Gamma(\mathscr{C})) \end{bmatrix} \\ &= \begin{bmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \bigvee \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}. \end{aligned}$$

By Definition 2.5, we have that

$$\mathcal{X}_{SH(X)} = \Gamma(\mathscr{C}^+) \cdot \mathcal{X}_X = \begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}^T;$$

$$\mathcal{X}_{SL(X)} = \Gamma(\mathscr{C}^+) \odot \mathcal{X}_X = \begin{bmatrix} 1 & 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 \end{bmatrix} \odot \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}^T.$$

Therefore, $SH(X) = \{x_1, x_2, x_3, x_4, x_5, x_6\}$ *and* $SL(X) = \{x_3\}$.

In Example 3.3, we only need to calculate elements in $\triangle_1(\Gamma(\mathscr{C}))$, $\triangle_2(\Gamma(\mathscr{C}))$ and $\triangle_3(\Gamma(\mathscr{C}))$ by Theorem 3.2. Thereby, the incremental algorithm is effective to compute the second lower and upper approximations of sets.

In practical situations, there exists a need to construct the type-2 characteristic matrices of dynamic coverings for computing the sixth lower and upper approximations of sets. Subsequently, we construct $\prod(\mathscr{C}^+)$ based on $\prod(\mathscr{C})$. For convenience, we denote $\prod(\mathscr{C}) = (d_{ij})_{n\times n}$ and $\prod(\mathscr{C}^+) = (e_{ij})_{(n+t)\times(n+t)}$.

Theorem 3.4 *Let* $(U^+, \mathscr{C}^+)$ *be a dynamic covering approximation space of* $(U, \mathscr{C})$, $\prod(\mathscr{C})$ *and* $\prod(\mathscr{C}^+)$ *the*

type-2 characteristic matrices of $\mathscr{C}$ and $\mathscr{C}^+$, respectively. Then

$$\prod(\mathscr{C}^+) = \begin{bmatrix} \prod(\mathscr{C}) & 1 \\ 1 & 1 \end{bmatrix} \bigwedge \begin{bmatrix} \triangle_1(\prod(\mathscr{C})) & \triangle_3(\prod(\mathscr{C})) \\ \triangle_2(\prod(\mathscr{C})) & \triangle_4(\prod(\mathscr{C})) \end{bmatrix},$$

where

$$\begin{aligned}
\triangle_1(\prod(\mathscr{C})) &= \begin{bmatrix} a_{1(m+1)} & a_{2(m+1)} & . & . & . & a_{n(m+1)} \\ a_{1(m+2)} & a_{2(m+2)} & . & . & . & a_{n(m+2)} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{1(m+l)} & a_{2(m+l)} & . & . & . & a_{n(m+l)} \end{bmatrix}^T \odot \begin{bmatrix} a_{1(m+1)} & a_{2(m+1)} & . & . & . & a_{n(m+1)} \\ a_{1(m+2)} & a_{2(m+2)} & . & . & . & a_{n(m+2)} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{1(m+l)} & a_{2(m+l)} & . & . & . & a_{n(m+l)} \end{bmatrix}; \\
\triangle_2(\prod(\mathscr{C})) &= \begin{bmatrix} a_{(n+1)1} & a_{(n+1)2} & . & . & . & a_{(n+1)(m+l)} \\ a_{(n+2)1} & a_{(n+2)2} & . & . & . & a_{(n+2)(m+l)} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{(n+t)1} & a_{(n+t)2} & . & . & . & a_{(n+t)(m+l)} \end{bmatrix} \odot \begin{bmatrix} a_{11} & a_{12} & . & . & . & a_{1(m+l)} \\ a_{21} & a_{22} & . & . & . & a_{2(m+l)} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{n1} & a_{n2} & . & . & . & a_{n(m+l)} \end{bmatrix}^T; \\
\triangle_3(\prod(\mathscr{C})) &= \begin{bmatrix} a_{11} & a_{12} & . & . & . & a_{1(m+l)} \\ a_{21} & a_{22} & . & . & . & a_{2(m+l)} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{n1} & a_{n2} & . & . & . & a_{n(m+l)} \end{bmatrix} \odot \begin{bmatrix} a_{(n+1)1} & a_{(n+1)2} & . & . & . & a_{(n+1)(m+l)} \\ a_{(n+2)1} & a_{(n+2)2} & . & . & . & a_{(n+2)(m+l)} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{(n+t)1} & a_{(n+t)2} & . & . & . & a_{(n+t)(m+l)} \end{bmatrix}^T; \\
\triangle_4(\prod(\mathscr{C})) &= \begin{bmatrix} a_{(n+1)1} & a_{(n+1)2} & . & . & . & a_{(n+1)(m+l)} \\ a_{(n+2)1} & a_{(n+2)2} & . & . & . & a_{(n+2)(m+l)} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{(n+t)1} & a_{(n+t)2} & . & . & . & a_{(n+t)(m+l)} \end{bmatrix} \odot \begin{bmatrix} a_{(n+1)1} & a_{(n+1)2} & . & . & . & a_{(n+1)(m+l)} \\ a_{(n+2)1} & a_{(n+2)2} & . & . & . & a_{(n+2)(m+l)} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{(n+t)1} & a_{(n+t)2} & . & . & . & a_{(n+t)(m+l)} \end{bmatrix}^T.
\end{aligned}$$

Proof. By Definition 3.1, we have $\prod(\mathscr{C})$ and $\prod(\mathscr{C}^+)$ as follows:

$$\begin{aligned}
\prod(\mathscr{C}) &= M_{\mathscr{C}} \odot M_{\mathscr{C}}^T \\
&= \begin{bmatrix} a_{11} & a_{12} & . & . & . & a_{1m} \\ a_{21} & a_{22} & . & . & . & a_{2m} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{n1} & a_{n2} & . & . & . & a_{nm} \end{bmatrix} \odot \begin{bmatrix} a_{11} & a_{12} & . & . & . & a_{1m} \\ a_{21} & a_{22} & . & . & . & a_{2m} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{n1} & a_{n2} & . & . & . & a_{nm} \end{bmatrix}^T \\
&= \begin{bmatrix} d_{11} & d_{12} & . & . & . & d_{1n} \\ d_{21} & d_{22} & . & . & . & d_{2n} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ d_{n1} & d_{n2} & . & . & . & d_{nn} \end{bmatrix};
\end{aligned}$$

$$
\begin{aligned}
\prod(\mathscr{C}^+) &= M_{\mathscr{C}^+} \odot M_{\mathscr{C}^+}^T \\
&= \begin{bmatrix}
a_{11} & a_{12} & \cdots & a_{1m} & a_{1(m+1)} & \cdots & a_{1(m+l)} \\
a_{21} & a_{22} & \cdots & a_{2m} & a_{2(m+1)} & \cdots & a_{2(m+l)} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
a_{n1} & a_{n2} & \cdots & a_{nm} & a_{n(m+1)} & \cdots & a_{n(m+l)} \\
a_{(n+1)1} & a_{(n+1)2} & \cdots & a_{(n+1)m} & a_{(n+1)(m+1)} & \cdots & a_{(n+1)(m+l)} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
a_{(n+t)1} & a_{(n+t)2} & \cdots & a_{(n+t)m} & a_{(n+t)(m+1)} & \cdots & a_{(n+t)(m+l)}
\end{bmatrix} \\
&\quad \odot \begin{bmatrix}
a_{11} & a_{12} & \cdots & a_{1m} & a_{1(m+1)} & \cdots & a_{1(m+l)} \\
a_{21} & a_{22} & \cdots & a_{2m} & a_{2(m+1)} & \cdots & a_{2(m+l)} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
a_{n1} & a_{n2} & \cdots & a_{nm} & a_{n(m+1)} & \cdots & a_{n(m+l)} \\
a_{(n+1)1} & a_{(n+1)2} & \cdots & a_{(n+1)m} & a_{(n+1)(m+1)} & \cdots & a_{(n+1)(m+l)} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
a_{(n+t)1} & a_{(n+t)2} & \cdots & a_{(n+t)m} & a_{(n+t)(m+1)} & \cdots & a_{(n+t)(m+l)}
\end{bmatrix}^T \\
&= \begin{bmatrix}
e_{11} & e_{12} & \cdots & e_{1n} & e_{1(n+1)} & \cdots & e_{1(n+t)} \\
e_{21} & e_{22} & \cdots & e_{2n} & e_{2(n+1)} & \cdots & e_{1(n+t)} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
e_{n1} & e_{n2} & \cdots & e_{nn} & e_{n(n+1)} & \cdots & e_{1(n+t)} \\
e_{(n+1)1} & e_{(n+1)2} & \cdots & e_{(n+1)n} & e_{(n+1)(n+1)} & \cdots & e_{1(n+1)} \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
e_{(n+t)1} & e_{(n+t)2} & \cdots & e_{(n+t)n} & e_{(n+t)(n+1)} & \cdots & e_{(n+t)(n+t)}
\end{bmatrix}.
\end{aligned}
$$

In the sense of the type-2 characteristic matrice of $\mathscr{C}^+$, we have

$$
\begin{aligned}
e_{11} &= [\ a_{11} \ \ a_{12} \ \cdots \ a_{1m} \ \ a_{1(m+1)} \ \cdots \ a_{1(m+l)}\ ] \\
&\quad \odot [\ a_{11} \ \ a_{12} \ \cdots \ a_{1m} \ \ a_{1(m+1)} \ \cdots \ a_{1(m+l)}\ ]^T \\
&= [\ a_{11} \ \ a_{12} \ \cdots \ a_{1m}\ ] \odot [\ a_{11} \ \ a_{12} \ \cdots \ a_{1m}\ ]^T \\
&\quad \wedge [\ a_{1(m+1)} \ \cdots \ a_{1(m+l)}\ ] \odot [\ a_{1(m+1)} \ \cdots \ a_{1(m+l)}\ ]^T \\
&= d_{11} \wedge [\ a_{1(m+1)} \ \cdots \ a_{1(m+l)}\ ] \odot [\ a_{1(m+1)} \ \cdots \ a_{1(m+l)}\ ]^T; \\
e_{(n+1)1} &= [\ a_{(n+1)1} \ \ a_{(n+1)2} \ \cdots \ a_{(n+1)m} \ \ a_{(n+1)(m+1)} \ \cdots \ a_{(n+1)(m+l)}\ ] \\
&\quad \odot [\ a_{11} \ \ a_{12} \ \cdots \ a_{1m} \ \ a_{1(m+1)} \ \cdots \ a_{1(m+l)}\ ]^T \\
&= 1 \wedge [\ a_{(n+1)1} \ \ a_{(n+1)2} \ \cdots \ a_{(n+1)m} \ \ a_{(n+1)(m+1)} \ \cdots \ a_{(n+1)(m+l)}\ ] \\
&\quad \odot [\ a_{11} \ \ a_{12} \ \cdots \ a_{1m} \ \ a_{1(m+1)} \ \cdots \ a_{1(m+l)}\ ]^T;
\end{aligned}
$$

$$\begin{aligned}
e_{1(n+1)} &= [\ a_{11}\ \ a_{12}\ \ .\ .\ .\ \ a_{1m}\ \ a_{1(m+1)}\ \ .\ .\ .\ \ a_{1(m+l)}\ ] \\
&\quad \odot [\ a_{(n+1)1}\ \ a_{(n+1)2}\ \ .\ .\ .\ \ a_{(n+1)m}\ \ a_{(n+1)(m+1)}\ \ .\ .\ .\ \ a_{(n+1)(m+l)}\ ]^T \\
&= 1 \wedge [\ a_{11}\ \ a_{12}\ \ .\ .\ .\ \ a_{1m}\ \ a_{1(m+1)}\ \ .\ .\ .\ \ a_{1(m+l)}\ ] \\
&\quad \odot [\ a_{(n+1)1}\ \ a_{(n+1)2}\ \ .\ .\ .\ \ a_{(n+1)m}\ \ a_{(n+1)(m+1)}\ \ .\ .\ .\ \ a_{(n+1)(m+l)}\ ]^T; \\
e_{(n+1)(n+1)} &= [\ a_{(n+1)1}\ \ a_{(n+1)2}\ \ .\ .\ .\ \ a_{(n+1)m}\ \ a_{(n+1)(m+1)}\ \ .\ .\ .\ \ a_{(n+1)(m+l)}\ ] \\
&\quad \odot [\ a_{(n+1)1}\ \ a_{(n+1)2}\ \ .\ .\ .\ \ a_{(n+1)m}\ \ a_{(n+1)(m+1)}\ \ .\ .\ .\ \ a_{(n+1)(m+l)}\ ]^T \\
&= 1 \wedge [\ a_{(n+1)1}\ \ a_{(n+1)2}\ \ .\ .\ .\ \ a_{(n+1)m}\ \ a_{(n+1)(m+1)}\ \ .\ .\ .\ \ a_{(n+1)(m+l)}\ ] \\
&\quad \odot [\ a_{(n+1)1}\ \ a_{(n+1)2}\ \ .\ .\ .\ \ a_{(n+1)m}\ \ a_{(n+1)(m+1)}\ \ .\ .\ .\ \ a_{(n+1)(m+l)}\ ]^T.
\end{aligned}$$

Since $e_{11} \in \triangle_1(\prod(\mathscr{C}))$, $e_{(n+1)1} \in \triangle_2(\prod(\mathscr{C}))$, $e_{1(n+1)} \in \triangle_3(\prod(\mathscr{C}))$ and $e_{(n+1)(n+1)} \in \triangle_3(\prod(\mathscr{C}))$, we can compute other elements of $\triangle_1(\prod(\mathscr{C}))$, $\triangle_2(\prod(\mathscr{C}))$, $\triangle_3(\prod(\mathscr{C}))$ and $\triangle_4(\prod(\mathscr{C}))$ similarly. Thus, to compute $\prod(\mathscr{C}^+)$ on the basis of $\prod(\mathscr{C})$, we only need to compute $\triangle_1(\prod(\mathscr{C}))$, $\triangle_2(\prod(\mathscr{C}))$, $\triangle_3(\prod(\mathscr{C}))$ and $\triangle_4(\prod(\mathscr{C}))$ as follows:

$$\triangle_1(\prod(\mathscr{C})) = \begin{bmatrix} a_{1(m+1)} & a_{2(m+1)} & . & . & . & a_{n(m+1)} \\ a_{1(m+2)} & a_{2(m+2)} & . & . & . & a_{n(m+2)} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{1(m+l)} & a_{2(m+l)} & . & . & . & a_{n(m+l)} \end{bmatrix}^T \odot \begin{bmatrix} a_{1(m+1)} & a_{2(m+1)} & . & . & . & a_{n(m+1)} \\ a_{1(m+2)} & a_{2(m+2)} & . & . & . & a_{n(m+2)} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{1(m+l)} & a_{2(m+l)} & . & . & . & a_{n(m+l)} \end{bmatrix};$$

$$\triangle_2(\prod(\mathscr{C})) = \begin{bmatrix} a_{(n+1)1} & a_{(n+1)2} & . & . & . & a_{(n+1)(m+l)} \\ a_{(n+2)1} & a_{(n+2)2} & . & . & . & a_{(n+2)(m+l)} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{(n+t)1} & a_{(n+t)2} & . & . & . & a_{(n+t)(m+l)} \end{bmatrix} \odot \begin{bmatrix} a_{11} & a_{12} & . & . & . & a_{1(m+l)} \\ a_{21} & a_{22} & . & . & . & a_{2(m+l)} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{n1} & a_{n2} & . & . & . & a_{n(m+l)} \end{bmatrix}^T;$$

$$\triangle_3(\prod(\mathscr{C})) = \begin{bmatrix} a_{11} & a_{12} & . & . & . & a_{1(m+l)} \\ a_{21} & a_{22} & . & . & . & a_{2(m+l)} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{n1} & a_{n2} & . & . & . & a_{n(m+l)} \end{bmatrix} \odot \begin{bmatrix} a_{(n+1)1} & a_{(n+1)2} & . & . & . & a_{(n+1)(m+l)} \\ a_{(n+2)1} & a_{(n+2)2} & . & . & . & a_{(n+2)(m+l)} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{(n+t)1} & a_{(n+t)2} & . & . & . & a_{(n+t)(m+l)} \end{bmatrix}^T;$$

$$\triangle_4(\prod(\mathscr{C})) = \begin{bmatrix} a_{(n+1)1} & a_{(n+1)2} & . & . & . & a_{(n+1)(m+l)} \\ a_{(n+2)1} & a_{(n+2)2} & . & . & . & a_{(n+2)(m+l)} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{(n+t)1} & a_{(n+t)2} & . & . & . & a_{(n+t)(m+l)} \end{bmatrix} \odot \begin{bmatrix} a_{(n+1)1} & a_{(n+1)2} & . & . & . & a_{(n+1)(m+l)} \\ a_{(n+2)1} & a_{(n+2)2} & . & . & . & a_{(n+2)(m+l)} \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ a_{(n+t)1} & a_{(n+t)2} & . & . & . & a_{(n+t)(m+l)} \end{bmatrix}^T.$$

Therefore, we have

$$\prod(\mathscr{C}^+) = \begin{bmatrix} \prod(\mathscr{C}) & 1 \\ 1 & 1 \end{bmatrix} \bigwedge \begin{bmatrix} \triangle_1(\prod(\mathscr{C})) & \triangle_3(\prod(\mathscr{C})) \\ \triangle_2(\prod(\mathscr{C})) & \triangle_4(\prod(\mathscr{C})) \end{bmatrix}.$$

The following example illustrates that how to compute the sixth lower and upper approximations of set by using the incremental algorithm.

Example 3.5 *(Continuation of Example 3.3) We obtain that*

$$\prod(\mathscr{C}) = M_{\mathscr{C}} \odot M_{\mathscr{C}}^T = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix} \odot \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}^T = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

By Theorem 3.4, we have that

$$\triangle_1(\prod(\mathscr{C})) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}^T \odot \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix};$$

$$\triangle_2(\prod(\mathscr{C})) = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix} \odot \begin{bmatrix} 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 \end{bmatrix}^T = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix};$$

$$\triangle_3(\prod(\mathscr{C})) = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 \end{bmatrix}^T \odot \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix};$$

$$\triangle_4(\prod(\mathscr{C})) = \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix} \odot \begin{bmatrix} 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 \end{bmatrix}^T = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Thus, we have

$$\prod(\mathscr{C}^+) = \begin{bmatrix} \prod(\mathscr{C}) & 1 \\ 1 & 1 \end{bmatrix} \bigwedge \begin{bmatrix} \triangle_1(\prod(\mathscr{C})) & \triangle_3(\prod(\mathscr{C})) \\ \triangle_2(\prod(\mathscr{C})) & \triangle_4(\prod(\mathscr{C})) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

By Definition 2.5, we obtain

$$\mathcal{X}_{XH(X)} = \prod(\mathscr{C}^+) \cdot \mathcal{X}_X = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}^T;$$

$$\mathcal{X}_{XL(X)} = \prod(\mathscr{C}^+) \odot \mathcal{X}_X = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \odot \begin{bmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 0 \end{bmatrix}^T.$$

Therefore, $XH(X) = \{x_2, x_3, x_4, x_5\}$ and $XL(X) = \{x_3, x_4, x_5\}$.

In Example 3.5, we need to compute all elements in $\prod(\mathscr{C}^+)$ for constructing approximations of sets by Definition 3.1. By Theorem 3.4, we only need to calculate elements in $\triangle_1(\prod(\mathscr{C}))$, $\triangle_2(\prod(\mathscr{C}))$, $\triangle_3(\prod(\mathscr{C}))$ and $\triangle_4(\prod(\mathscr{C}))$. Thereby, the incremental algorithm is more effective to compute approximations of sets.

4 Non-incremental and incremental algorithms for computing the second and sixth lower and upper approximations of sets

In this section, we show non-incremental and incremental algorithms of computing the second lower and upper approximations of sets.

Algorithm 4.1 *(Non-incremental algorithm of computing $SH_{\mathscr{C}^+}(X^+)$ and $SL_{\mathscr{C}^+}(X^+)$(**NCS**))*

Step 1: Input $(U^+, \mathscr{C}^+)$ and $X^+ \subseteq U^+$;

Step 2: Construct $M_{\mathscr{C}^+}$ and $\Gamma(\mathscr{C}^+) = M_{\mathscr{C}^+} \cdot M^T_{\mathscr{C}^+}$;

Step 3: Compute $\mathcal{X}_{SH_{\mathscr{C}^+}(X^+)} = \Gamma(\mathscr{C}^+) \cdot \mathcal{X}_{X^+}$ and $\mathcal{X}_{SL_{\mathscr{C}^+}(X^+)} = \Gamma(\mathscr{C}^+) \odot \mathcal{X}_{X^+}$;

Step 4: Output $SH_{\mathscr{C}^+}(X^+)$ and $SL_{\mathscr{C}^+}(X^+)$.

Algorithm 4.2 *(Incremental algorithm of computing $SH_{\mathscr{C}^+}(X^+)$ and $SL_{\mathscr{C}^+}(X^+)$(**ICS**))*

Step 1: Input $(U, \mathscr{C})$, $(U^+, \mathscr{C}^+)$ and $X \subseteq U^+$;

Step 2: Calculate $\Gamma(\mathscr{C}) = M_{\mathscr{C}} \cdot M^T_{\mathscr{C}}$, where $M_{\mathscr{C}} = (a_{ij})_{n\times m}$;

Step 3: Compute $\triangle_1(\Gamma(\mathscr{C}))$ and $\triangle_2(\Gamma(\mathscr{C}))$ and $\triangle_3(\Gamma(\mathscr{C}))$;

Step 4: Construct $\Gamma(\mathscr{C}^+)$, where

$$\Gamma(\mathscr{C}^+) \quad = \quad (c_{ij})_{(n+1)(n+1)} = \begin{bmatrix} \Gamma(\mathscr{C}) & 0 \\ 0 & 0 \end{bmatrix} \bigvee \begin{bmatrix} \triangle_1(\Gamma(\mathscr{C})) & (\triangle_2(\Gamma(\mathscr{C})))^T \\ \triangle_2(\Gamma(\mathscr{C})) & \triangle_3(\Gamma(\mathscr{C})) \end{bmatrix};$$

Step 5: Obtain $\mathcal{X}_{SH(X)}$ and $\mathcal{X}_{SL(X)}$, where

$$\mathcal{X}_{SH(X)} \quad = \quad \Gamma(\mathscr{C}^+) \cdot \mathcal{X}_X; \mathcal{X}_{SL(X)} = \Gamma(\mathscr{C}^+) \odot \mathcal{X}_X.$$

Subsequently, we present non-incremental and incremental algorithms of computing the sixth lower and upper approximations of sets.

Algorithm 4.3 *(Non-incremental algorithm of computing $XH_{\mathscr{C}^+}(X^+)$ and $XL_{\mathscr{C}^+}(X^+)$(**NCX**))*

Step 1: Input $(U^+, \mathscr{C}^+)$ and $X^+ \subseteq U^+$;

Step 2: Construct $M_{\mathscr{C}^+}$ and $\prod(\mathscr{C}^+) = M_{\mathscr{C}^+} \cdot M^T_{\mathscr{C}^+}$;

Step 3: Compute $\mathcal{X}_{XH_{\mathscr{C}^+}(X^+)} = \prod(\mathscr{C}^+) \cdot \mathcal{X}_{X^+}$ and $\mathcal{X}_{XL_{\mathscr{C}^+}(X^+)} = \prod(\mathscr{C}^+) \odot \mathcal{X}_{X^+}$;

Step 4: Output $XH_{\mathscr{C}^+}(X^+)$ and $XL_{\mathscr{C}^+}(X^+)$.

Algorithm 4.4 *(Incremental algorithm of computing $XH_{\mathscr{C}^+}(X^+)$ and $XL_{\mathscr{C}^+}(X^+)$(**ICX**))*

Step 1: Input $(U, \mathscr{C})$, $(U^+, \mathscr{C}^+)$ and $X \subseteq U^+$;

Step 2: Construct $\prod(\mathscr{C})$, where $\prod(\mathscr{C}) = M_{\mathscr{C}} \odot M^T_{\mathscr{C}}$;

Step 3: Compute $\triangle_1(\prod(\mathscr{C}))$ and $\triangle_2(\prod(\mathscr{C}))$, $\triangle_3(\prod(\mathscr{C}))$ and $\triangle_4(\prod(\mathscr{C}))$;

Step 4: Calculate $\prod(\mathscr{C}^+)$, where $\prod(\mathscr{C}^+) = \begin{bmatrix} \prod(\mathscr{C}) & 1 \\ 1 & 1 \end{bmatrix} \bigwedge \begin{bmatrix} \triangle_1(\prod(\mathscr{C})) & \triangle_3(\prod(\mathscr{C})) \\ \triangle_2(\prod(\mathscr{C})) & \triangle_4(\prod(\mathscr{C})) \end{bmatrix}$;

Step 5: Get $\mathcal{X}_{XH(X)}$ and $\mathcal{X}_{XL(X)}$, where

$$\mathcal{X}_{XH(X)} = \prod(\mathscr{C}^{+}) \cdot \mathcal{X}_{X}; \mathcal{X}_{XL(X)} = \prod(\mathscr{C}^{+}) \odot \mathcal{X}_{X}.$$

5 Conclusions

In this paper, we have provided effective approaches to constructing approximations of concepts in dynamic covering approximation spaces. Concretely, we have constructed type-1 and type-2 characteristic matrices of coverings with the incremental approaches. Incremental algorithms have been presented for computing the second and sixth lower and upper approximations of sets. Several examples have been employed to illustrate that computing approximations of sets could be reduced greatly by using the incremental approaches.

In the future, we will propose more effective approaches to constructing the type-1 and type-2 characteristic matrices of coverings. Additionally, we will focus on the development of effective approaches for knowledge discovery in dynamic covering approximation spaces.

Acknowledgments

We would like to thank the anonymous reviewers very much for their professional comments and valuable suggestions. This work is supported by the National Natural Science Foundation of China (NO. 11201490,11371130,11401052,11401195), the Scientific Research Fund of Hunan Provincial Education Department(No.14C0049).

References

[1] S. Asharaf, M. Murty, S. Shevade, Rough set based incremental clustering of interval data, Pattern Recognition Letters 27(9) (2006) 515-519.

[2] H.M. Chen, T.R. Li, S.J. Qiao, D. Ruan, A rough set based dynamic maintenance approach for approximations in coarsening and refining attribute values, International Journal of Intelligent Systems 25(10) (2010) 1005-1026.

[3] H.M. Chen, T.R. Li, D. Ruan, Maintenance of approximations in incomplete ordered decision systems while attribute values coarsening or refining, Knowledge-Based Systems 31 (2012) 140-161.

[4] H.M. Chen, T.R. Li, D. Ruan, J.H. Lin, C.X. Hu, A rough-set based incremental approach for updating approximations under dynamic maintenance environments, IEEE Transactions on Knowledge and Data Engineering 25(2) (2013) 174-184.

[5] D.G. Chen, C.Z. Wang, Q.H. Hu, A new approach to attributes reduction of consistent and inconsistent covering decision systems with covering rough sets, Information Sciences 177(17) (2007) 3500-3518.

[6] Y. Du, Q.H. Hu, P.F. Zhu, P.J. Ma, Rule Learning for Classification Based on Neighborhood Covering Reduction, Information Sciences 181(24) (2011) 5457-5467.

[7] R. Elwell, R. Polikar, Incremental Learning of Concept drift in nonstationary environments, IEEE Transactions on Neural Networks 22(10) (2011) 1517-1531.

[8] Y.N. Fan, T.L. Tseng, C.C. Chen, C.C. Huang, Rule Induction Based On An Incremental Rough Set, Expert Systems With Application 36(9) (2009) 11439-11450.

[9] A. Frank, A. Asuncion, UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science (2010).

[10] Q.H. Hu, D.R. Yu, Z.X. Xie, Neighborhood classifiers, Expert Systems with Applications 34(2) (2008) 866-876.

[11] C.C. Huang, T.L. Tseng, Y.N. Fan, C.H. Hsu, Alternative rule induction methods based on incremental object using rough set theory, Applied Soft Computing 13 (2013) 372-389.

[12] F. Jiang, Y.F. Sui, C.G. Cao, An incremental decision tree algorithm based on rough sets and its application in intrusion detection, Artificial Intelligence Review 40 (2013) 517-530.

[13] P. Joshi, P. Kulkarni, Incremental Learning: Areas and Methods-A Survey, International Journal of Data Mining & Knowledge Management Process 2(5) (2012) 43-51.

[14] Y. Leung, W.Z. Wu, W.X. Zhang, Knowledge acquisition in incomplete information systems: a rough set approach, European Journal of Operational Research 168 (2006) 164-180.

[15] S.Y. Li, T.R. Li, D. Liu, Incremental updating approximations in dominance-based rough sets approach under the variation of the attribute set, Knowledge-Based Systems 40 (2013) 17-26.

[16] S.Y. Li, T.R. Li, D. Liu, Dynamic Maintenance of Approximations in Dominance-Based Rough Set Approach under the Variation of the Object Set, International Journal of Intelligent Systems 28(8) (2013) 729-751.

[17] J.H. Li, C.L. Mei, Y.J. Lv, A heuristic knowledge-reduction method for decision formal contexts, Computers and Mathematics with Applications 61 (2011) 1096-1106.

[18] J.H. Li, C.L. Mei, Y.J. Lv, Knowledge reduction in real decision formal contexts, Information Sciences 189 (2012) 191-207.

[19] J.H. Li, C.L. Mei, Y.J. Lv, Incomplete decision contexts: Approximate concept construction, rule acquisition and knowledge reduction, International Journal of Approximate Reasoning 54(1) (2013) 149-165.

[20] T.R. Li, D. Ruan, W. Geert, J. Song, Y. Xu, A rough sets based characteristic relation approach for dynamic attribute generalization in data mining, Knowledge-Based Systems 20(5) (2007) 485-494.

[21] T.R. Li, D. Ruan, J. Song, Dynamic maintenance of decision rules with rough set under characteristic relation, Wireless Communications, Networking and Mobile Computing (2007) 3713-3716.

[22] J.Y. Liang, F. Wang, C.Y. Dang, Y.H. Qian, A Group Incremental Approach to Feature Selection Applying Rough Set Technique, IEEE Transactions on Knowledge and Data Engineering 26(2) (2014) 294-308.

[23] D. Liu, T.R. Li, D. Ruan, J.B. Zhang, Incremental learning optimization on knowledge discovery in dynamic business intelligent systems, Journal of Global Optimization 51(2) (2011) 325-344.

[24] D. Liu, T.R. Li, D. Ruan, W.L. Zou, An incremental approach for inducing knowledge from dynamic information systems, Fundamenta Informaticae 94(2) (2009) 245-260.

[25] D. Liu, T.R. Li, J.B.Zhang, A rough set-based incremental approach for learning knowledge in dynamic incomplete information systems, International Journal of Approximate Reasoning 55(8) (2014) 1764-1786.

[26] C. Luo, T.R. Li, H.M. Chen, Dynamic maintenance of approximations in set-valued ordered decision systems under the attribute generalization, Information Sciences 257 (2014) 210-228.

[27] C. Luo, T.R. Li, H.M. Chen, D. Liu, Incremental approaches for updating approximations in set-valued ordered information systems, Knowledge-Based Systems 50 (2013) 218-233.

[28] C. Luo, T.R. Li, H.M. Chen, L.X. Lu, Fast algorithms for computing rough approximations in 4 set-valued decision systems while updating criteria values, Information Sciences (2014) http://dx.doi.org/10.1016/j.ins.2014.12.029.

[29] S. Ozawa, S. Pang, N. Kasabov, Incremental Learning of chunk data for online pattern classification systems, IEEE Transactions on Neural Networks 19(6) (2008) 1061-1074.

[30] Z. Pawlak, Rough Sets: Theoretical Aspects of Reasoning About Data, Kluwer Academic Publishers, Dordrecht, MA, 1991.

[31] Y.H. Qian, J.Y. Liang, D.Y. Li, F. Wang, N.N. Ma, Approximation reduction in inconsistent incomplete decision tables, Knowledge-Based Systems 23(5) (2010) 427-433.

[32] M.W. Shao, Y. Leung, W.Z. Wu, Rule acquisition and complexity reduction in formal decision contexts, International Journal of Approximate Reasoning 55 (2014) 259-274.

[33] N. Shan, W. Ziarko, Data-based acquisition and incremental modification of classification rules, Computation Intelligence 11(2) (1995) 357-370.

[34] W.H. Shu, H. Shen, Updating attribute reduction in incomplete decision systems with the variation of attribute set, International Journal of Approximate Reasoning 55(3) (2013) 867-884.

[35] W.H. Shu, H. Shen, Incremental feature selection based on rough set in dynamic incomplete data, Pattern Recognition 47(12) (2014) 3890-3906.

[36] A. Skowron, C. Rauszer, The discernibility matrices and functions in information systems, in: R. Slowin' ski (Ed.), Intelligent Decision Support, Handbook of Applications and Advances of the Rough Sets Theory, Kluwer, Dordrecht, 1992.

[37] C.Z. Wang, M.W. Shao, B.Q. Sun, Q.H. Hu, An improved attribute reduction scheme with covering based rough sets, Applied Soft Computing 26 (2015) 235-243.

[38] C.Z. Wang, D.G. Chen, B.Q. Sun, Q.H. Hu, Communication between information systems with covering based rough sets, Information Sciences 216 (2012) 17-33.

[39] C.Z. Wang, Q. He, D.G. Chen, Q.H. Hu, A novel method for attribute reduction of covering decision systems, Information Sciences 254 (2014) 181-196.

[40] C.Z. Wang, D.G. Chen, C. Wu, Q.H. Hu, Data compression with homomorphism in covering information systems, International Journal of Approximate Reasoning 52 (2011) 519-525.

[41] F. Wang, J.Y. Liang, C.Y. Dang, Attribute reduction for dynamic data sets, Applied Soft Computing 13 (2013) 676-689.

[42] F. Wang, J.Y. Liang, Y.H. Qian, Attribute reduction: A dimension incremental strategy, Knowledge-Based Systems 39 (2013) 95-108.

[43] J. Wang, J. Wang, Reduction algorithms based on discernibility matrix: the ordered attributes method, Journal of Computer Science and Technology 16 (2001) 489-504.

[44] S.P. Wang, W. Zhu, Q.H. Zhu, F. Min, Characteristic matrix of covering and its application to boolean matrice decomposition and axiomatization, Information Sciences 263(1) (2014) 186-197.

[45] A. Wojna, Constraint Based Incremental Learning of Classification Rules, Lecture Notes in Computer Science 2005 (2001) 428-435.

[46] W.Z. Wu, W.X. Zhang, Neighborhood operator systems and approximations, Information Sciences 144(1) (2002) 201-217.

[47] W.H. Xu, W.X. Zhang, Measuring roughness of generalized rough sets induced by a covering, Fuzzy Sets and Systems 158(22) (2007) 2443-2455.

[48] Y.Y. Yao, Relational interpretations of neighborhood operators and rough set approximation operators, Information Sciences 111(1) (1998) 239-259.

[49] Y.Y. Yao, Y. Zhao, Discernibility matrix simplification for constructing attribute reducts, Information Sciences 179 (2009) 867-882.

[50] T. Yang, Q.G. Li, Reduction about approximation spaces of covering generalized rough sets, International Journal of Approximate Reasoning 51(3) (2010) 335-345.

[51] X.B. Yang, M. Zhang, H.L. Dou, J. Y. Yang, Neighborhood systems-based rough sets in incomplete information system, Knowledge-Based Systems 24(6) (2011) 858-867.

[52] X.B. Yang, Y. Qi, H.L. Yu, X.N. Song, J.Y. Yang, Updating multigranulation rough approximations with increasing of granular structures, Knowledge-Based Systems 64 (2014) 59-69.

[53] W. Zakowski, Approximations in the space (u, π), Demonstratio Mathematics 16 (1983) 761-769.

[54] J.B. Zhang, T.R. Li, D. Ruan, D. Liu, Rough sets based matric approaches with dynamic attribute variation in set-valued information systems, International Journal of Approximate Reasoning 53(4) (2012) 620-635.

[55] J.B. Zhang, T.R. Li, D. Ruan, D. Liu, Neighborhood rough sets for dynamic data mining, International Journal of Intelligent Systems 27(4) (2012) 317-342.

[56] J.B. Zhang, T.R. Li, H.M. Chen, Composite rough sets for dynamic data mining, Information Sciences 257 (2014) 81-100.

[57] Y.L. Zhang, J.J. Li, W.Z. Wu, On axiomatic characterizations of three pairs of covering based approximation operators, Information Sciences 180(2) (2010) 274-287.

[58] P. Zhu, Covering rough sets based on neighborhoods: an approach without using neighborhoods, International Journal of Approximate Reasoning 52(3) (2011) 461-472.

[59] W. Zhu, Topological approaches to covering rough sets, Information Sciences 177(6) (2007) 1499-1508.

[60] W. Zhu, Generalized rough sets based on relations, Information Sciences 177(22) (2007) 4997-5011.

[61] W. Zhu, Relationship among basic concepts in covering-based rough sets, Information Sciences 179(14) (2009) 2478-2486.

[62] W. Zhu, Relationship between generalized rough sets based on binary relation and coverings, Information Sciences 179(3) (2009) 210-225.